

FIG. 2

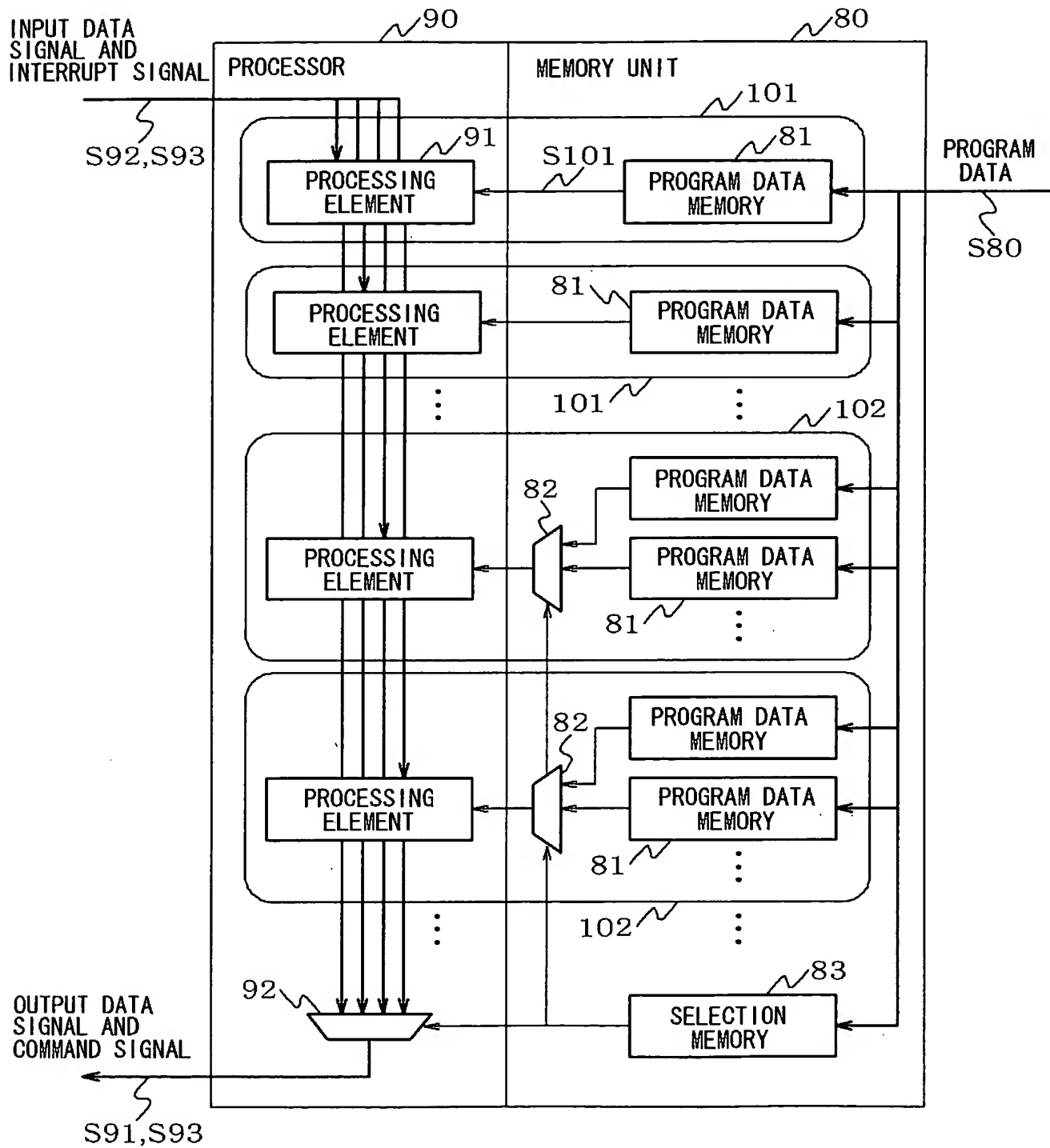


FIG . 3

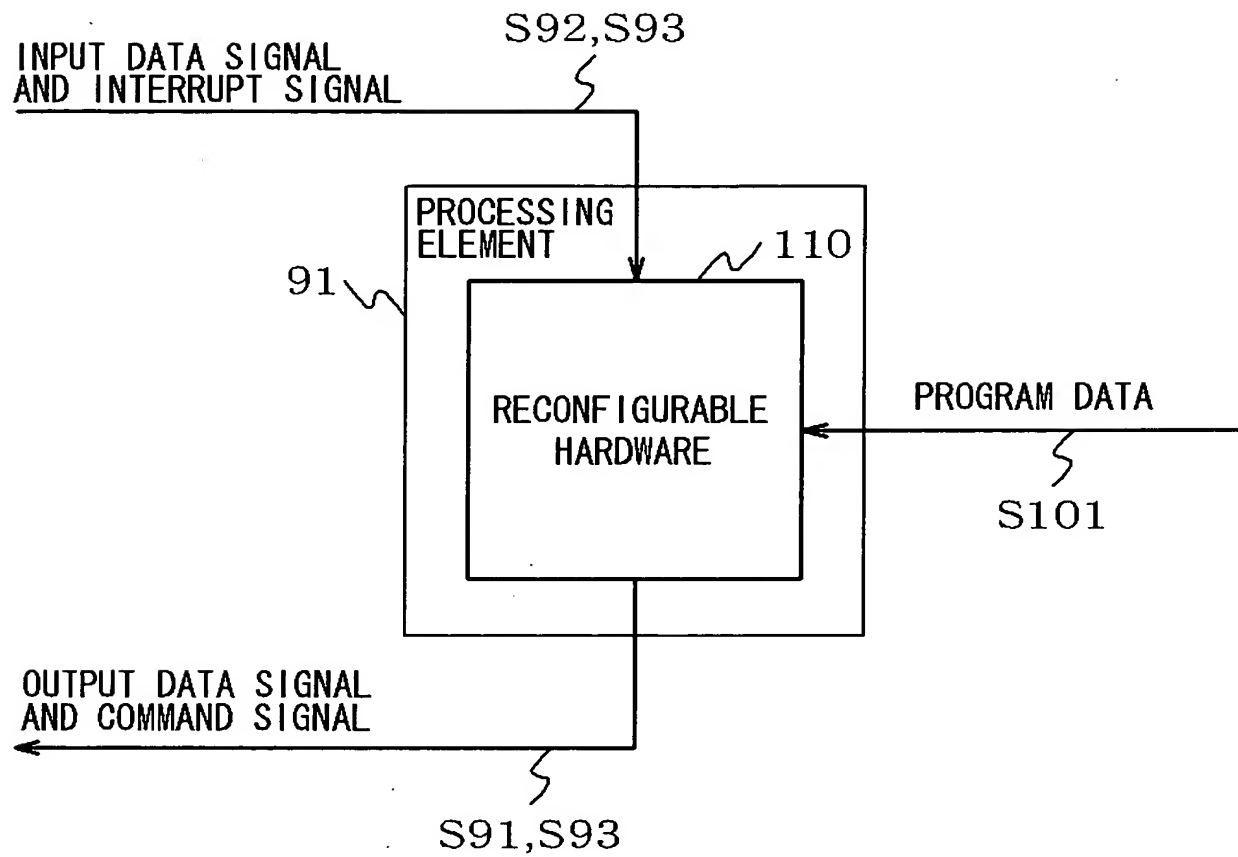


FIG . 4

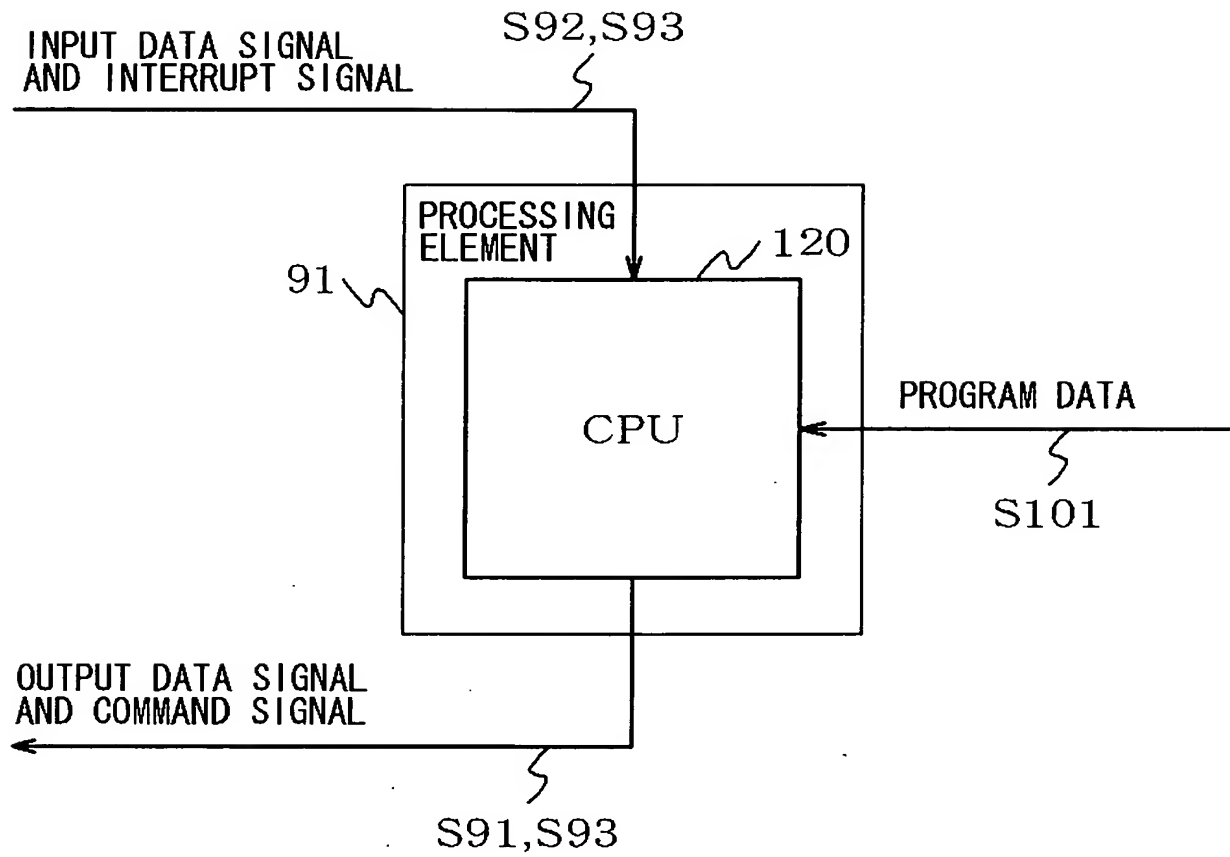


FIG. 5

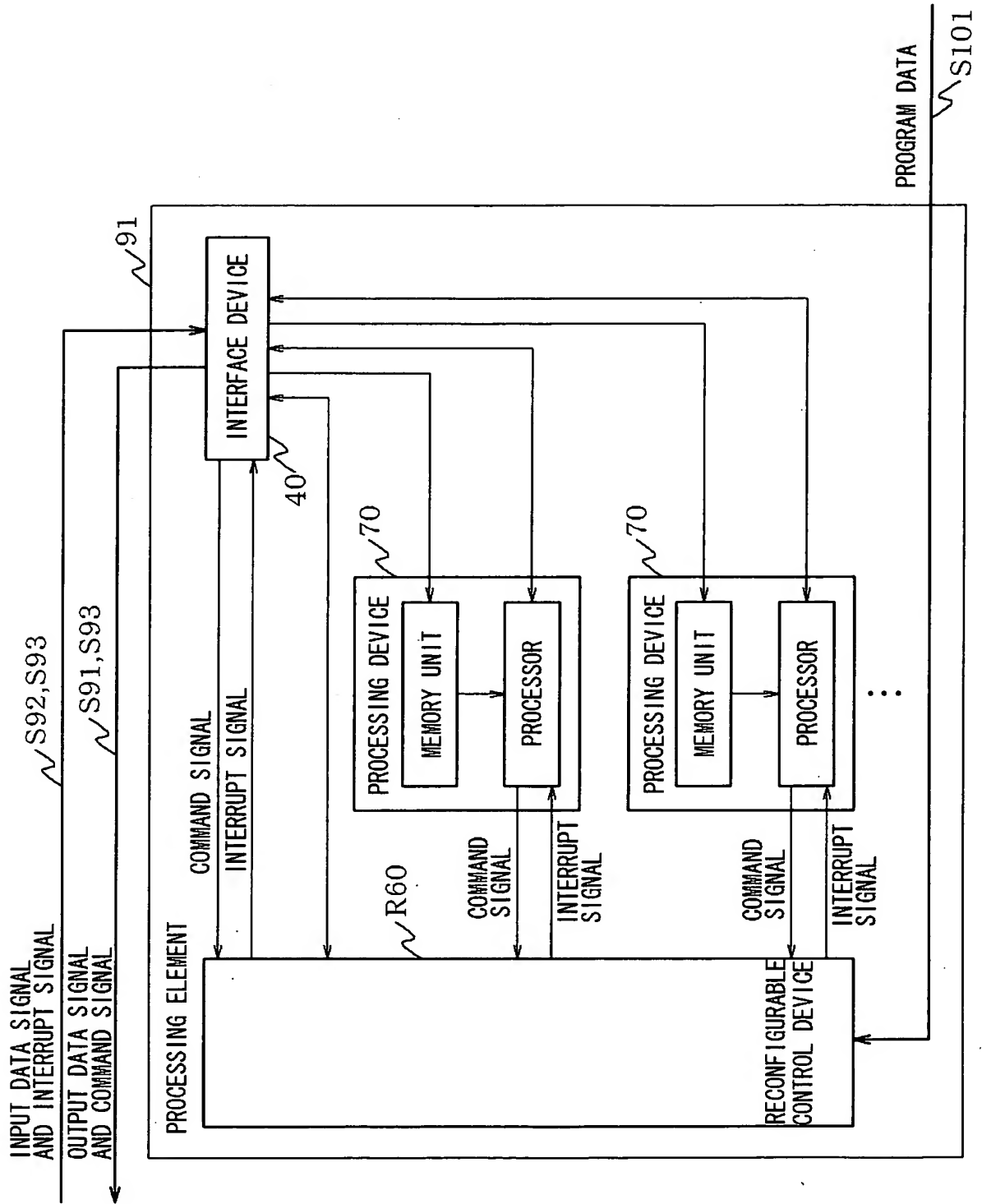


FIG . 6

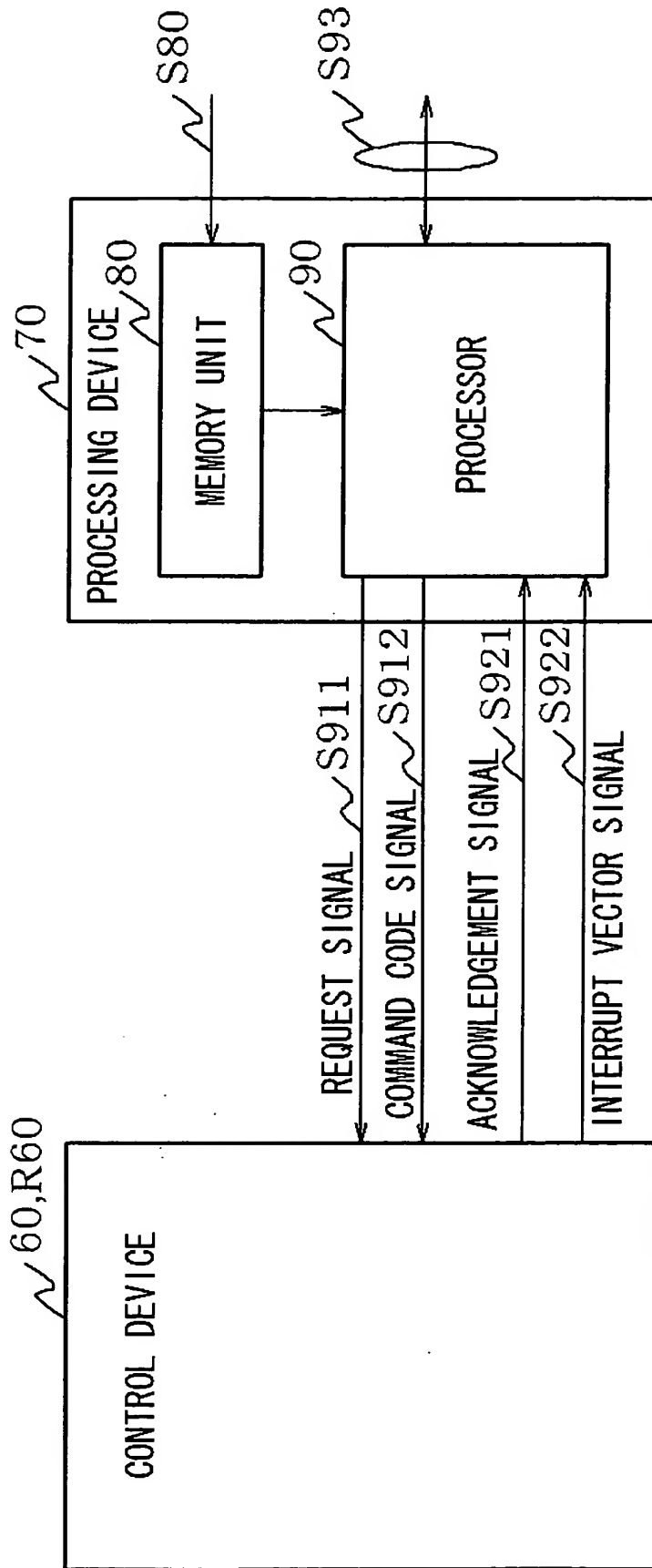


FIG . 7

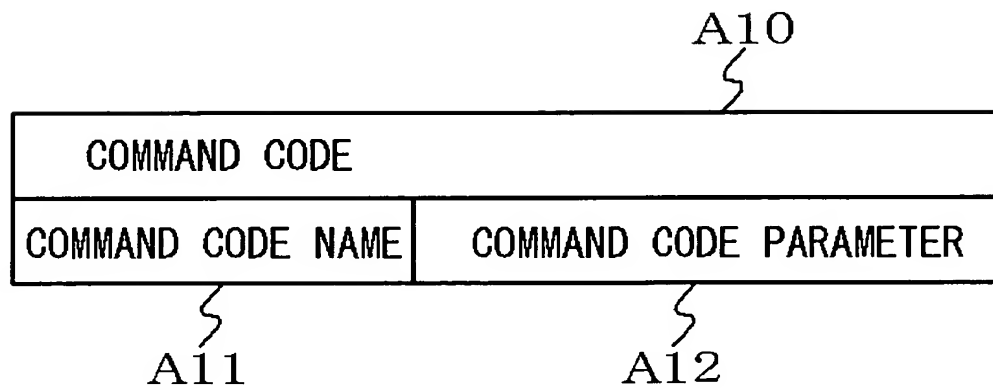


FIG. 8

COMMAND CODE NAME	COMMAND CODE PARAMETER	CONTENTS
activate	PROCESSING ELEMENT	SELECTS A SPECIFIED PROCESSING ELEMENT, STARTS A OPERATION
halt	PROCESSING DEVICE	HALTS A OPERATION OF A SPECIFIED PROCESSING DEVICE
Interrupt	PROCESSING DEVICE AND INTERRUPT VECTOR NUMBER	ISSUES A SPECIFIED INTERRUPT VECTOR NUMBER TO A SPECIFIED PROCESSING DEVICE
load_prg	REGION OF PROGRAM DATA MEMORY AND MEMORY REGION WHERE PROGRAM DATA IS STORED	TRANSFERS A PROGRAM DATA STORED IN A SPECIFIED MEMORY REGION TO A REGION OF A SPECIFIED PROGRAM DATA MEMORY
cancel_prg	REGION OF PROGRAM DATA MEMORY AND MEMORY REGION WHERE PROGRAM DATA IS STORED	CANCELS A TRANSFER OF A PROGRAM DATA STORED IN A SPECIFIED MEMORY REGION TO A REGION OF A SPECIFIED PROGRAM DATA MEMORY
wait_prg	REGION OF PROGRAM DATA MEMORY AND MEMORY REGION WHERE PROGRAM DATA IS STORED	WAITS UNTIL A TRANSFER OF A PROGRAM DATA STORED IN A SPECIFIED MEMORY REGION TO A REGION OF A SPECIFIED PROGRAM DATA MEMORY IS COMPLETE



FIG. 9

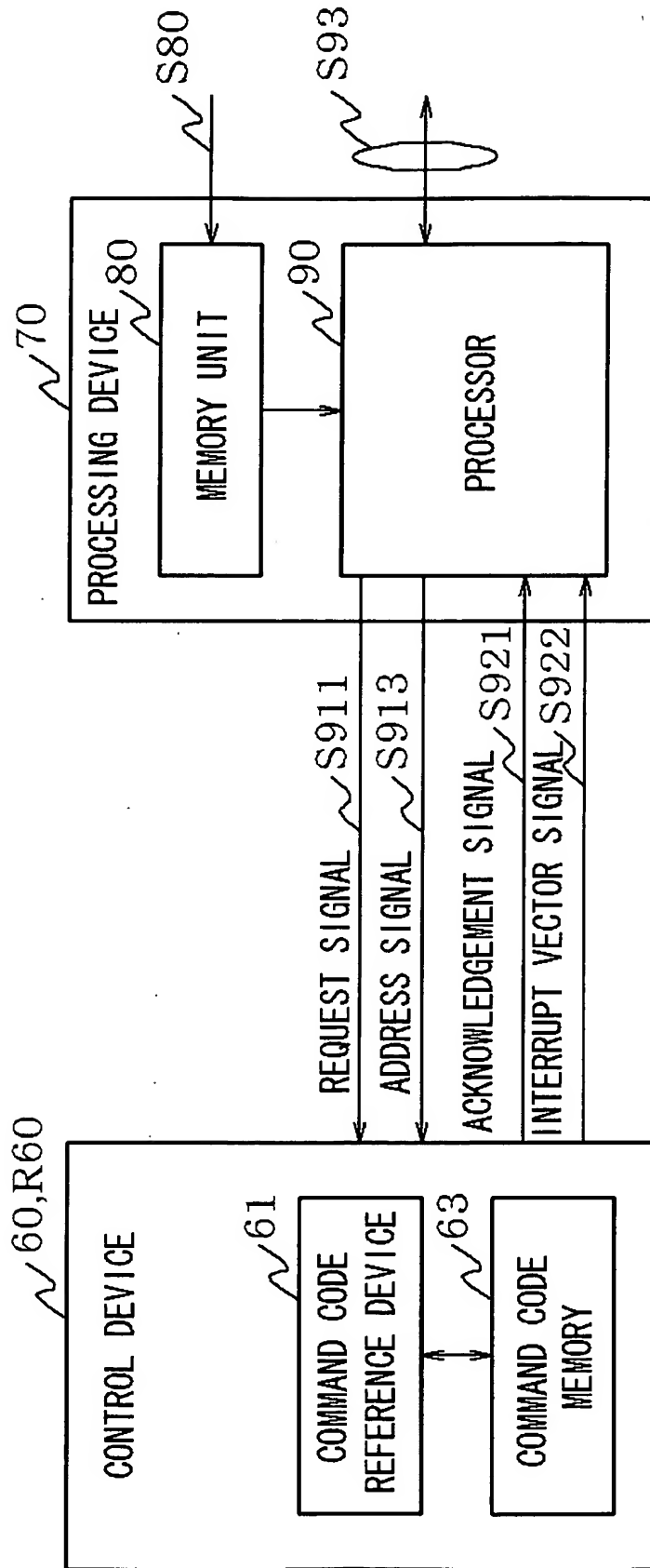


FIG . 10

ADDRESS	DATA
ADDRESS 1	COMMAND CODE 1
ADDRESS 2	COMMAND CODE 2
	<div>⋮</div> <div>A10</div>

FIG. 11

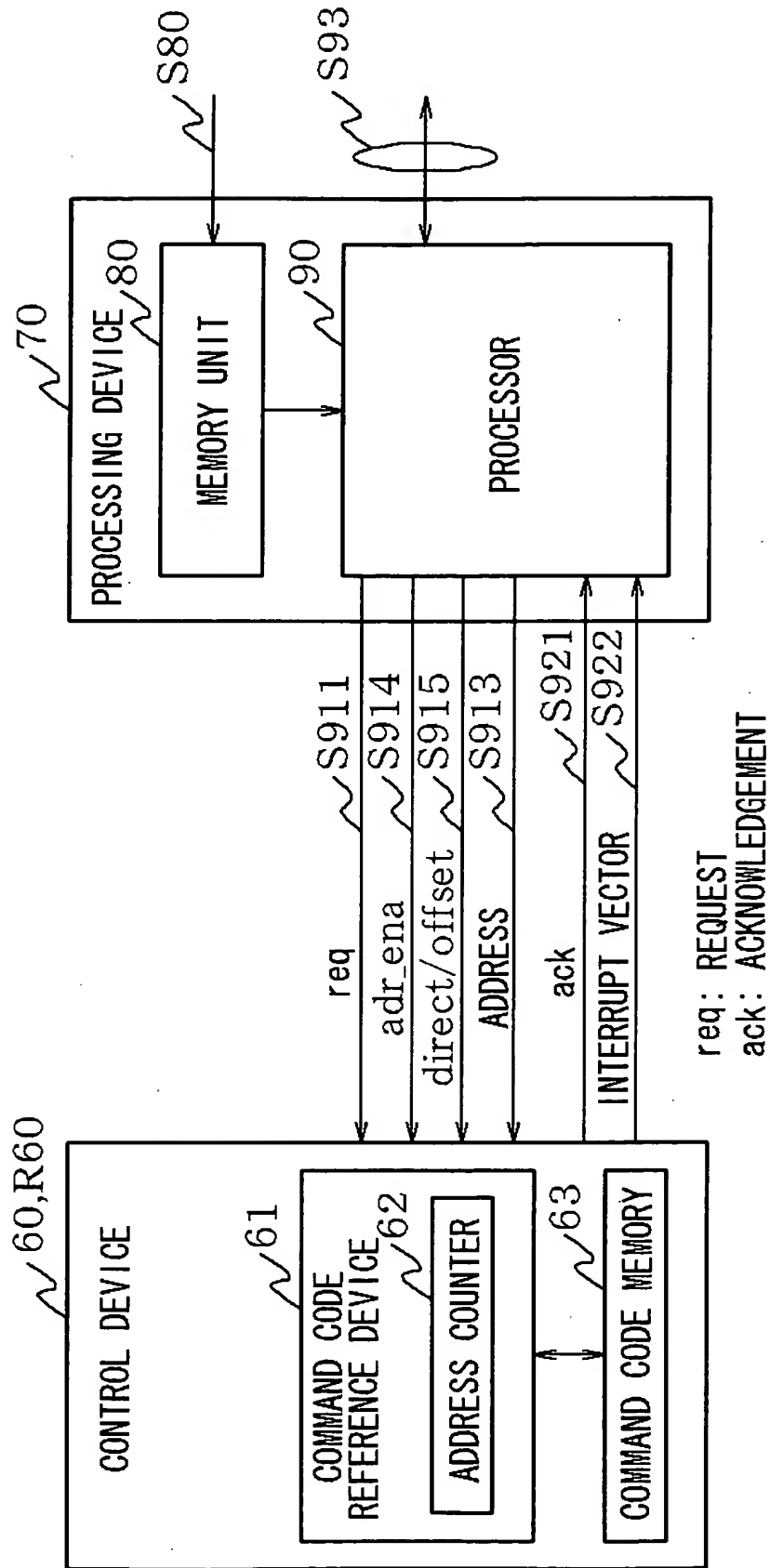


FIG . 12




ADDRESS	DATA		
ADDRESS 1	ADDRESS COUNTER CONTROL CODE 1	FLAG 1	COMMAND CODE 1
ADDRESS 2	ADDRESS COUNTER CONTROL CODE 2	FLAG 2	COMMAND CODE 2
	 A20	 A30	 A10

FIG . 13

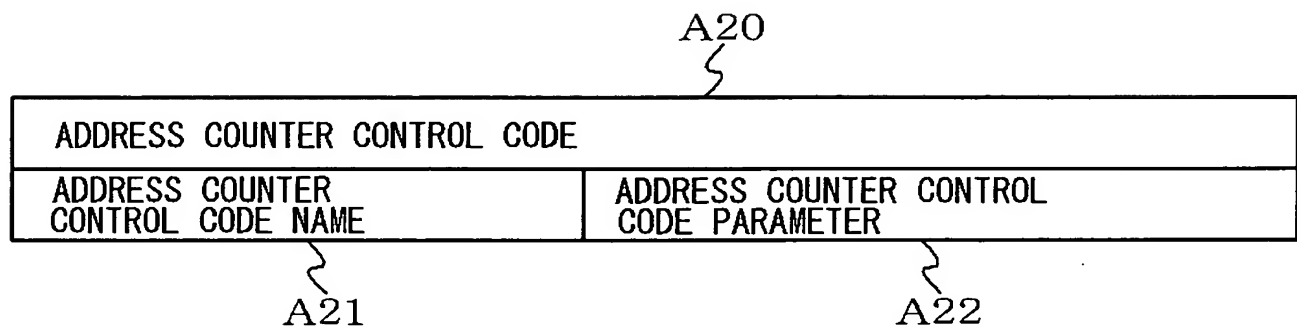


FIG. 14

ADDRESS COUNTER CONTROL CODE NAME	ADDRESS COUNTER CONTROL CODE PARAMETER	CONTENTS
load_adr	N	SETS N AS THE VALUE OF THE ADDRESS COUNTER
add_adr	N	ADDS N TO THE VALUE OF THE ADDRESS COUNTER
push_adr	N	HIDES THE VALUE OF THE ADDRESS COUNTER IN THE ADDRESS COUNTER STACK AND SETS N AS A NEW VALUE
pop_adr		RETURNS THE VALUE OF THE ADDRESS COUNTER FROM THE ADDRESS COUNTER STACK

N: NUMERICAL VALUE

FIG. 15







ADDRESS	DATA	
	ADDRESS COUNTER CONTROL CODE AND FLAG	COMMAND CODE
⋮	 X100           ⋮  Y100	
100	add_adr 1   cont	COMMAND CODE 100
101	load_adr 200   cont	COMMAND CODE 101
⋮	 X200           ⋮  X101  Y101  Y200	
200	add_adr 1   stop	COMMAND CODE 100
201	ADDRESS COUNTER CONTROL CODE 201	COMMAND CODE 101
⋮	⋮	

FIG. 16

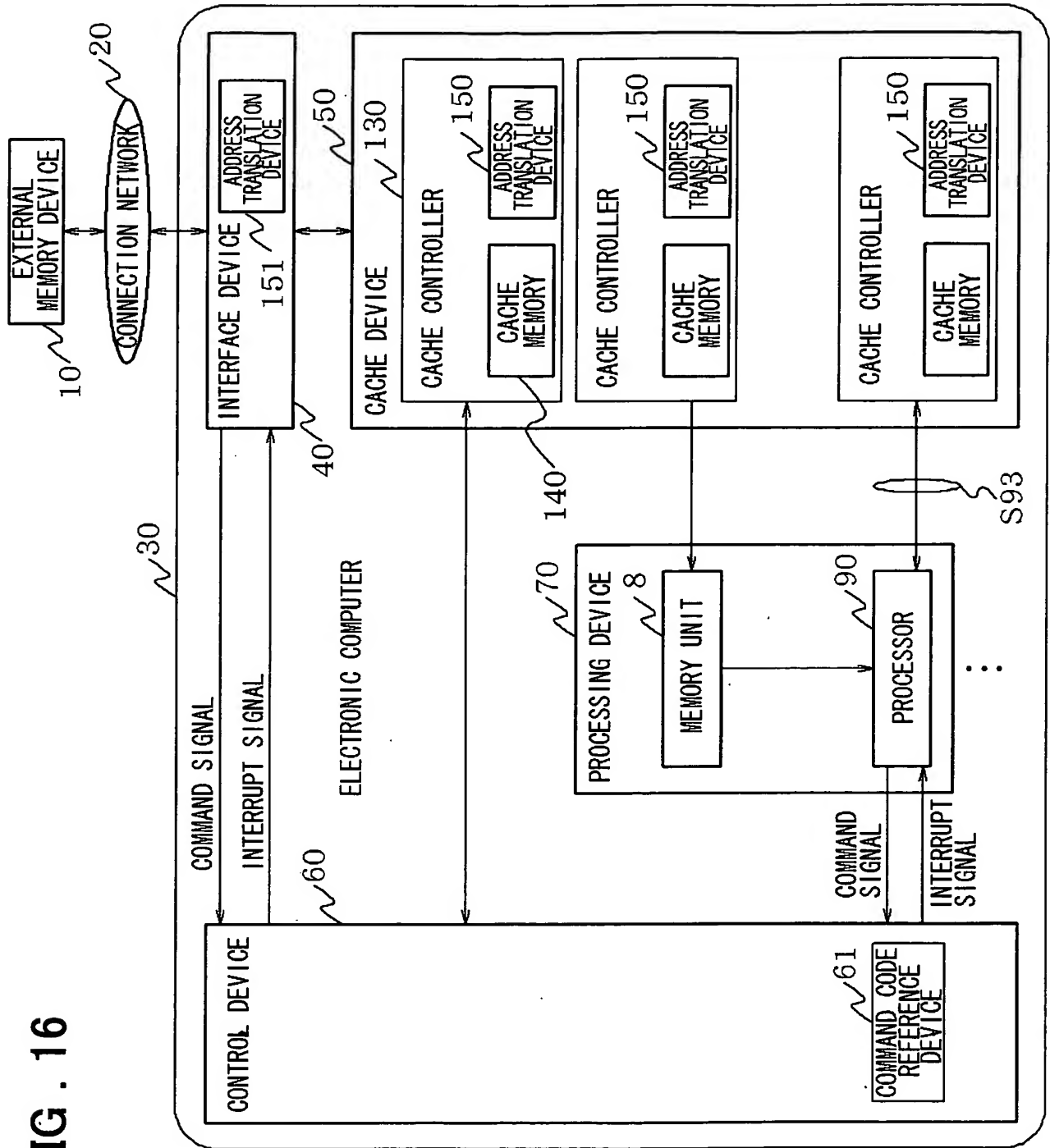




FIG. 17

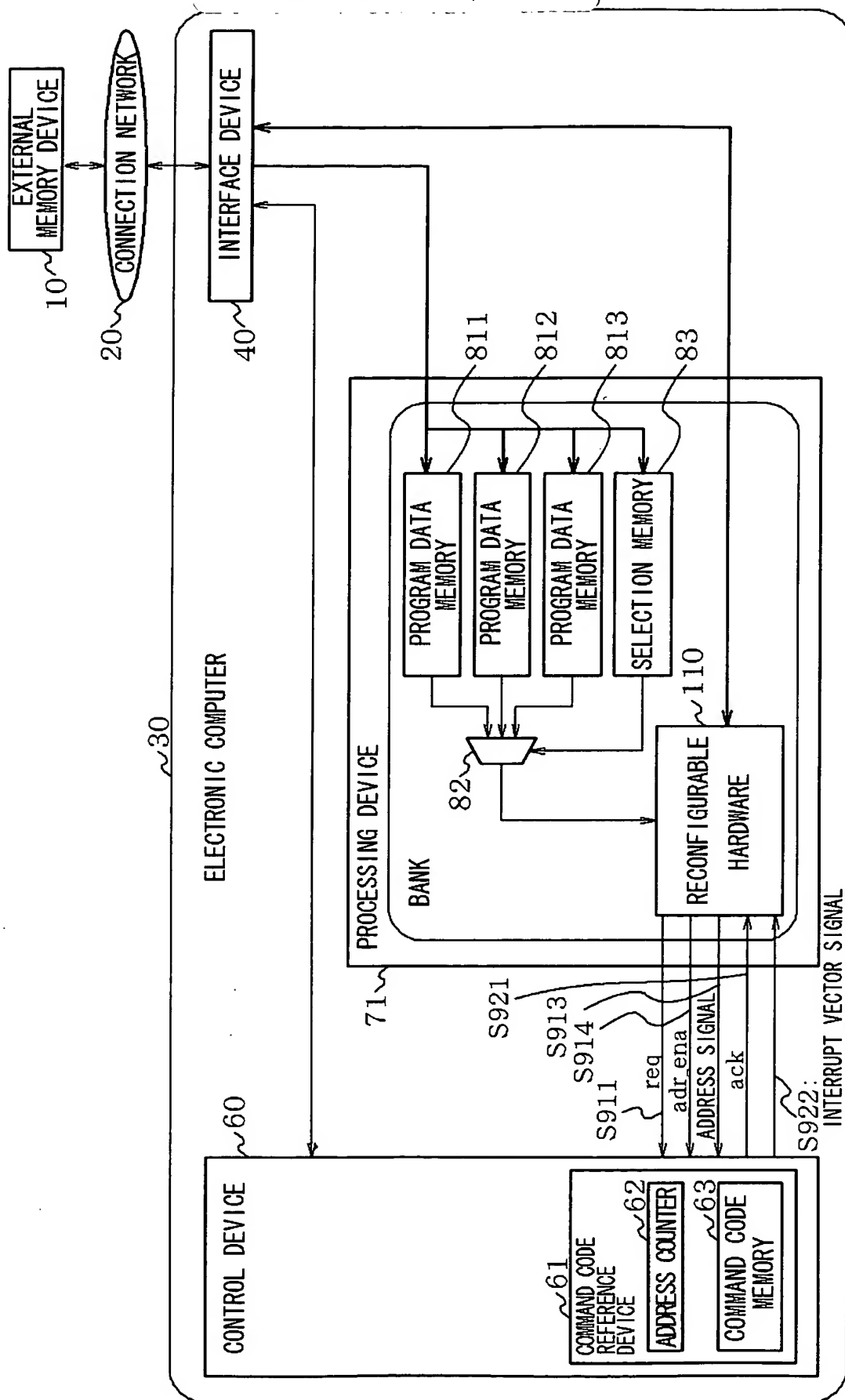


FIG . 18

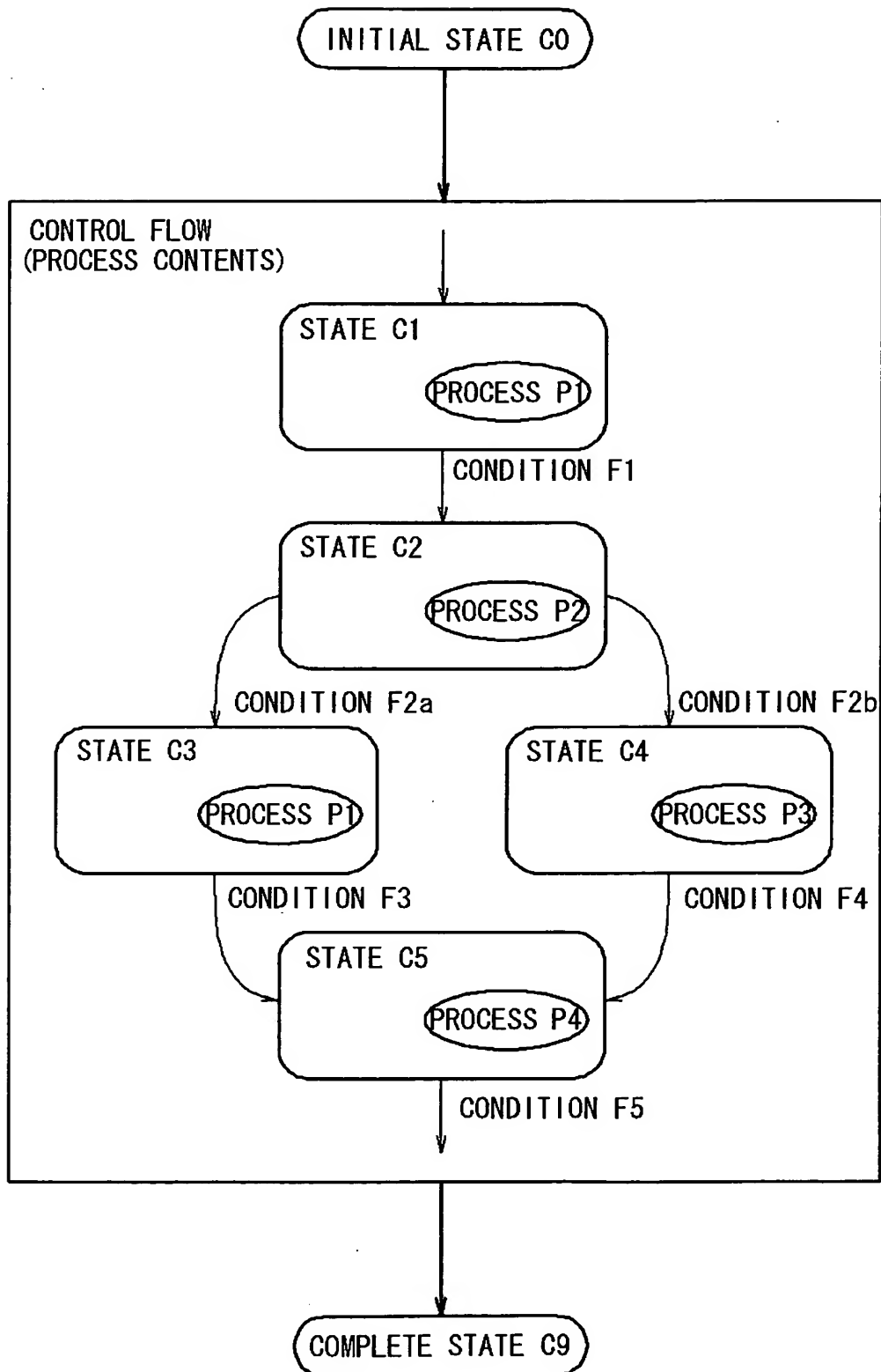


FIG . 19

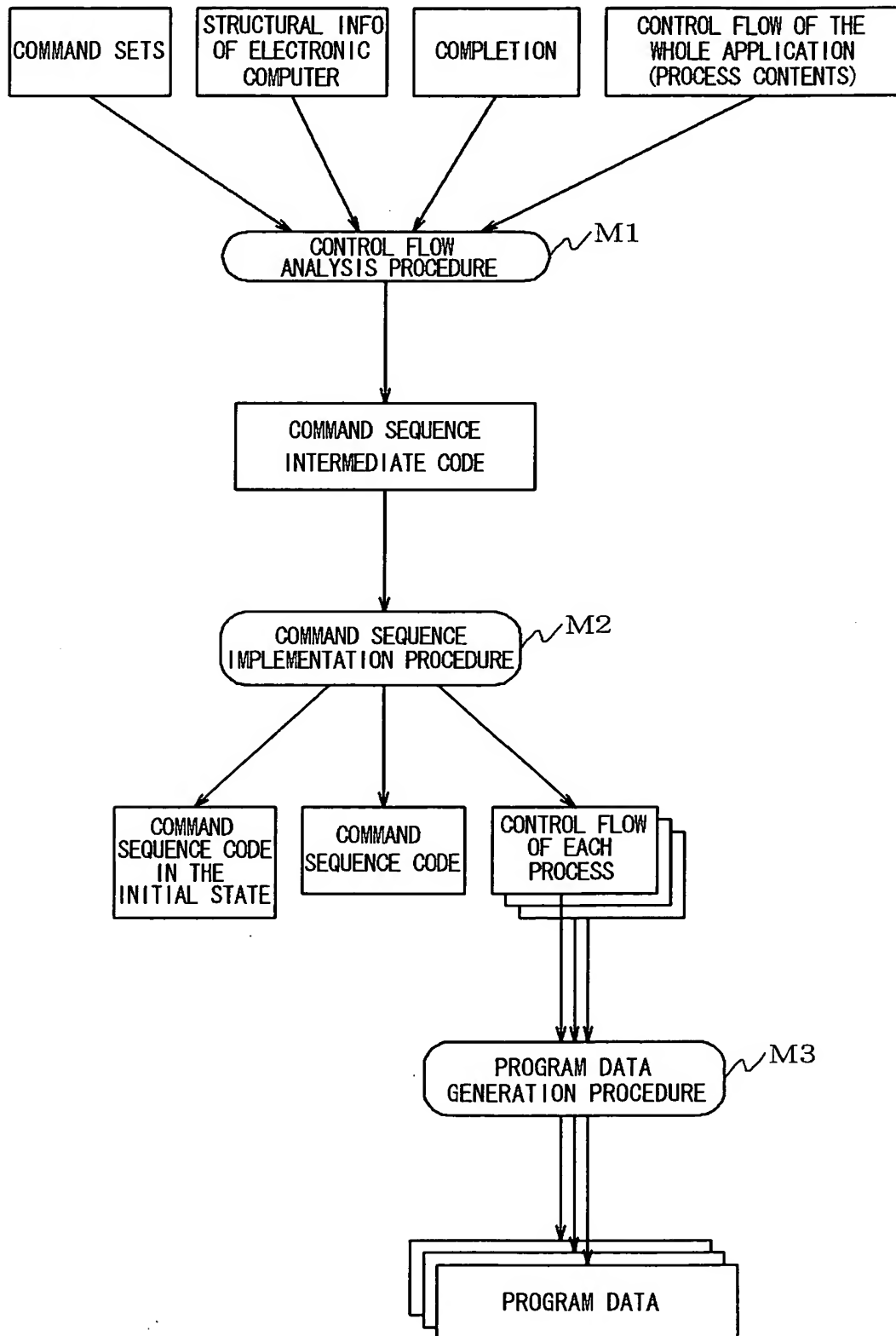


FIG. 20

PROCESS CONTENT	STATE	TRANSITION CONDITION	MOVES TO	PROGRAM DATA MEMORY USED	COMMAND SEQUENCE INTERMEDIATE CODE
—	INITIAL STATE C0	—	STATE C1	811	load_prg 812, PM1 wait_prg 812, PM1 interrupt 71, C1 activate 812 load_prg 813, PM2 SQ0
PROCESS P1	STATE C1	CONDITION F1 AND INTERRUPT VECTOR=C1	STATE C2	812	wait_prg 813, PM2 activate 813 load_prg 811, PM3 SQ1
	STATE C3	CONDITION F3 AND INTERRUPT VECTOR=C3	STATE C5		wait_prg 813, PM4 activate 813 SQ3
PROCESS P2	STATE C2	CONDITION F2a	STATE C3	813	cancel_prg 811, PM3 interrupt 71, C3 activate 812 load_prg 813, PM4 SQ2a
		CONDITION F2b	STATE C4		wait_prg 811, PM3 activate 811 load_prg 813, PM4 SQ2b
PROCESS P3	STATE C4	CONDITION F4	STATE C5	811	wait_prg 813, PM4 activate 813 SQ4
PROCESS P4	STATE C5	CONDITION F5	COMPLETION STATE C9	813	halt 71 SQ5

FIG. 21

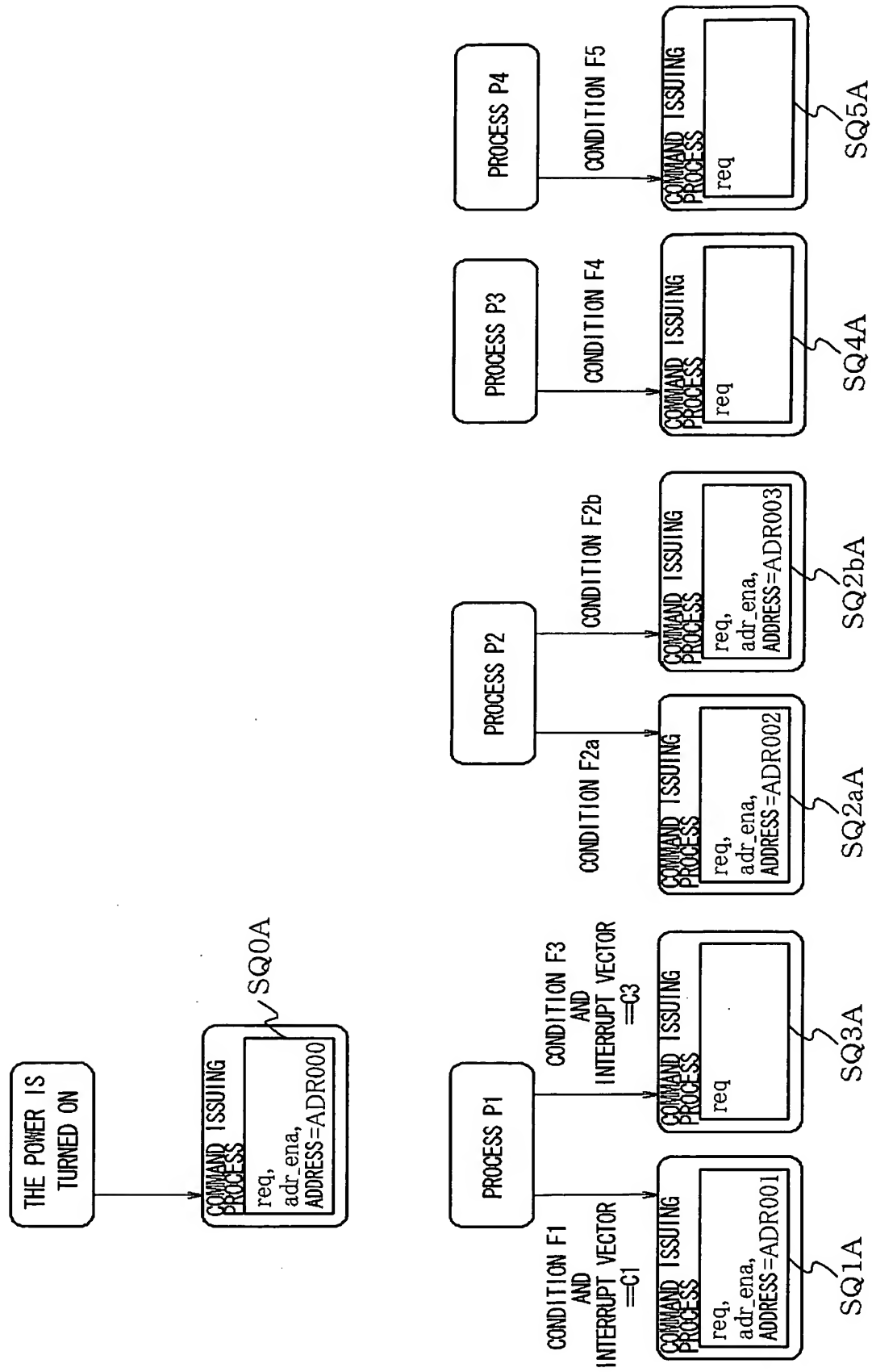


FIG. 22

BASE ADDRESS VALUE	OFFSET VALUE	COMMAND SEQUENCE		
		ADDRESS COUNTER CONTROL CODE	COMMAND CODE	
ADR000	+0	add_adr 1	cont	load_prg 812, PM1
	+1	add_adr 1	cont	wait_prg 812, PM1
	+2	add_adr 1	cont	interrupt 71, C1
	+3	add_adr 1	cont	activate 812
	+4	add_adr 0	stop	load_prg 813, PM2
ADR001	+0	add_adr 1	cont	wait_prg 813, PM2
	+1	add_adr 1	cont	activate 813
	+2	add_adr 0	stop	load_prg 811, PM3
ADR002	+0	add_adr 1	cont	cancel_prg 811, PM3
	+1	add_adr 1	cont	interrupt 71, C3
	+2	add_adr 1	cont	activate 812
	+3	load_adr ADR004	stop	load_prg 813, PM4
ADR003	+0	add_adr 1	cont	wait_prg 811, PM3
	+1	add_adr 1	cont	activate 811
	+2	load_adr ADR004	stop	load_prg 813, PM4
ADR004	+0	add_adr 1	cont	wait_prg 813, PM4
	+1	add_adr 1	stop	activate 813
	+2	add_adr 0	stop	halt 71

~ SQ0

~ SQ1

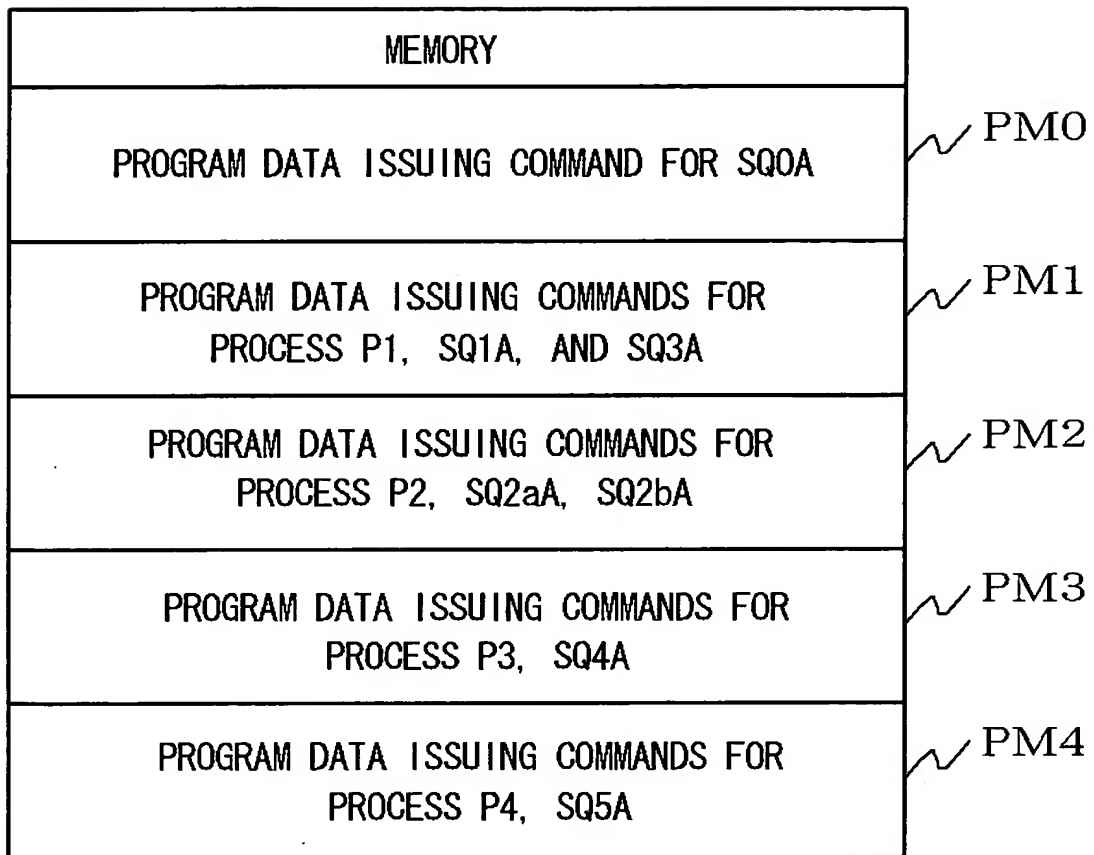
~ SQ2a

~ SQ2b

~ SQ3,SQ4

~ SQ5

FIG . 23



## FIG . 24

PMO IS STORED IN THE PROGRAM DATA MEMORY 811  
THE PROGRAM DATA MEMORY 811 IS ACTIVATED



FIG. 25

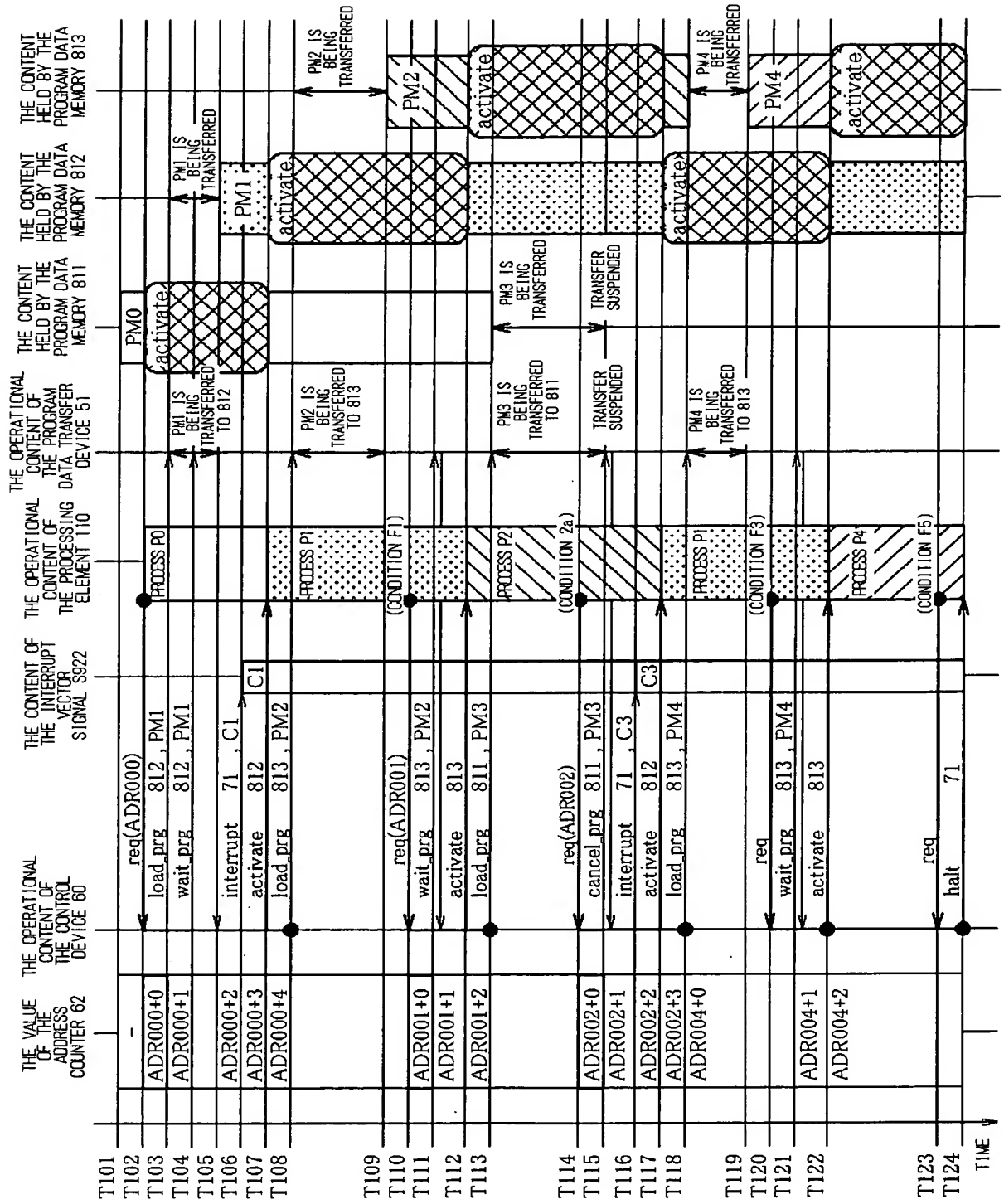


FIG. 26

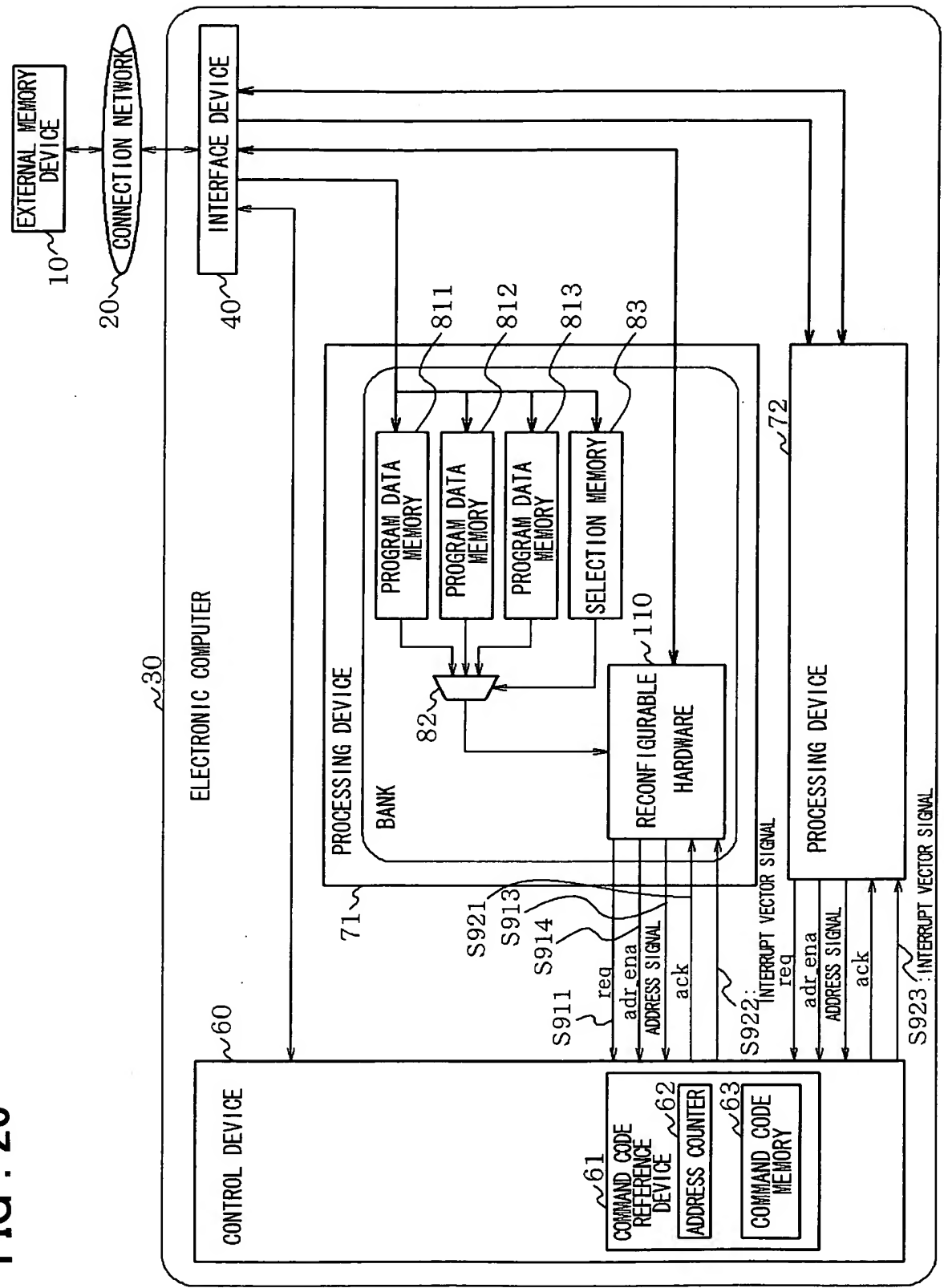


FIG. 27

BASE ADDRESS VALUE	OFFSET VALUE	COMMAND SEQUENCE		
		ADDRESS COUNTER CONTROL CODE	COMMAND CODE	
ADR000	+0	add_adr 1	cont	load_prg 812, PM1
	+1	add_adr 1	cont	wait_prg 812, PM1
	+2	add_adr 1	cont	interrupt 71, C1
	+3	add_adr 1	cont	activate 812
	+4	add_adr 0	stop	load_prg 813, PM2
ADR001	+0	add_adr 1	cont	wait_prg 813, PM2
	+1	add_adr 1	cont	activate 813
	+2	add_adr 0	stop	load_prg 811, PM3
ADR002	+0	add_adr 1	cont	cancel_prg 811, PM3
	+1	add_adr 1	cont	interrupt 71, C3
	+2	add_adr 1	cont	activate 812
	+3	load_adr ADR004	stop	load_prg 813, PM4
ADR003	+0	add_adr 1	cont	wait_prg 811, PM3
	+1	add_adr 1	cont	activate 811
	+2	load_adr ADR004	stop	load_prg 813, PM4
ADR004	+0	add_adr 1	cont	wait_prg 813, PM4
	+1	add_adr 1	stop	activate 813
	+2	add_adr 1	cont	halt 71
	+3	add_adr 0	stop	interrupt 72,END71

~ SQ0

~ SQ1

~ SQ2a

~ SQ2b

~ SQ3,SQ4

~ SQ5B

FIG . 28

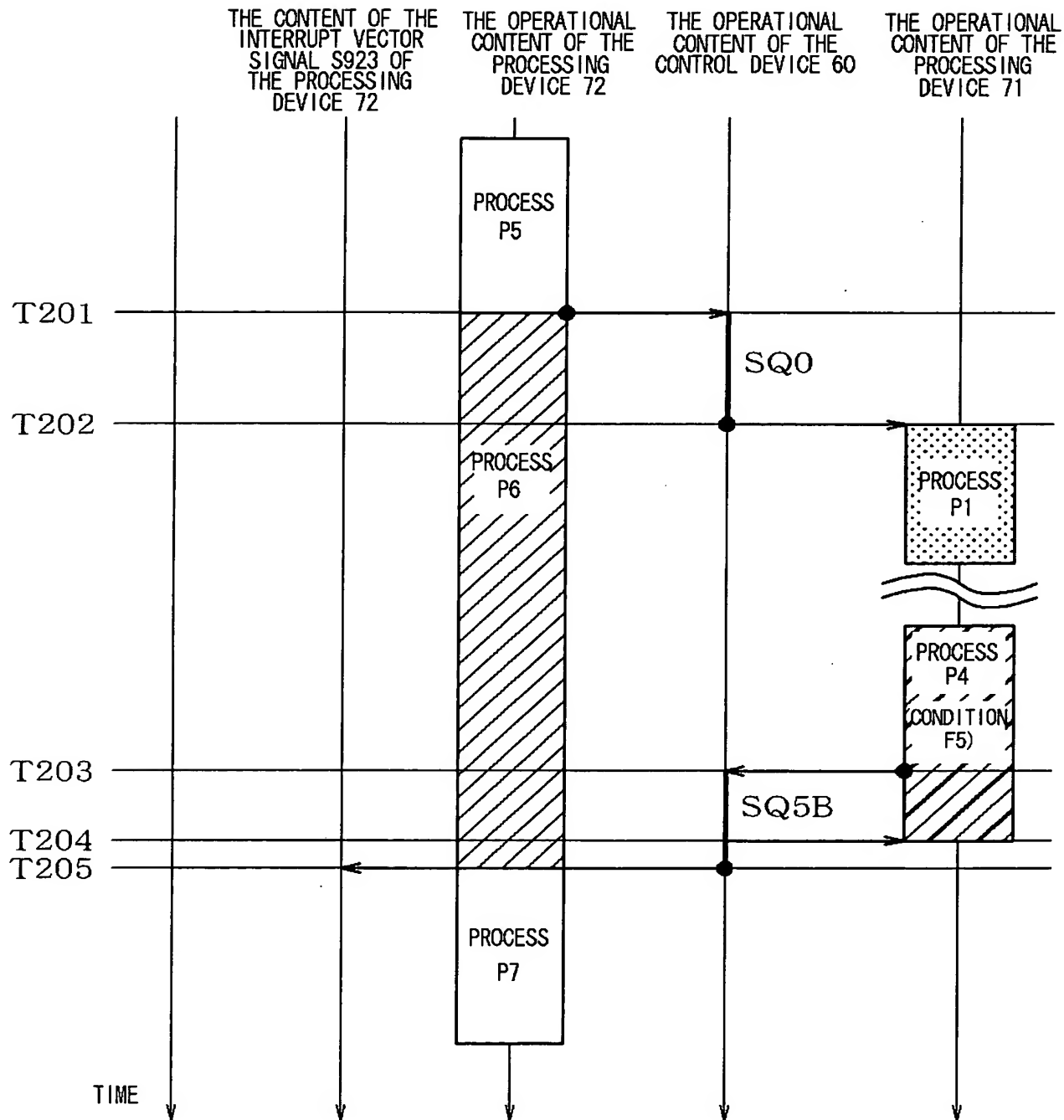


FIG . 29

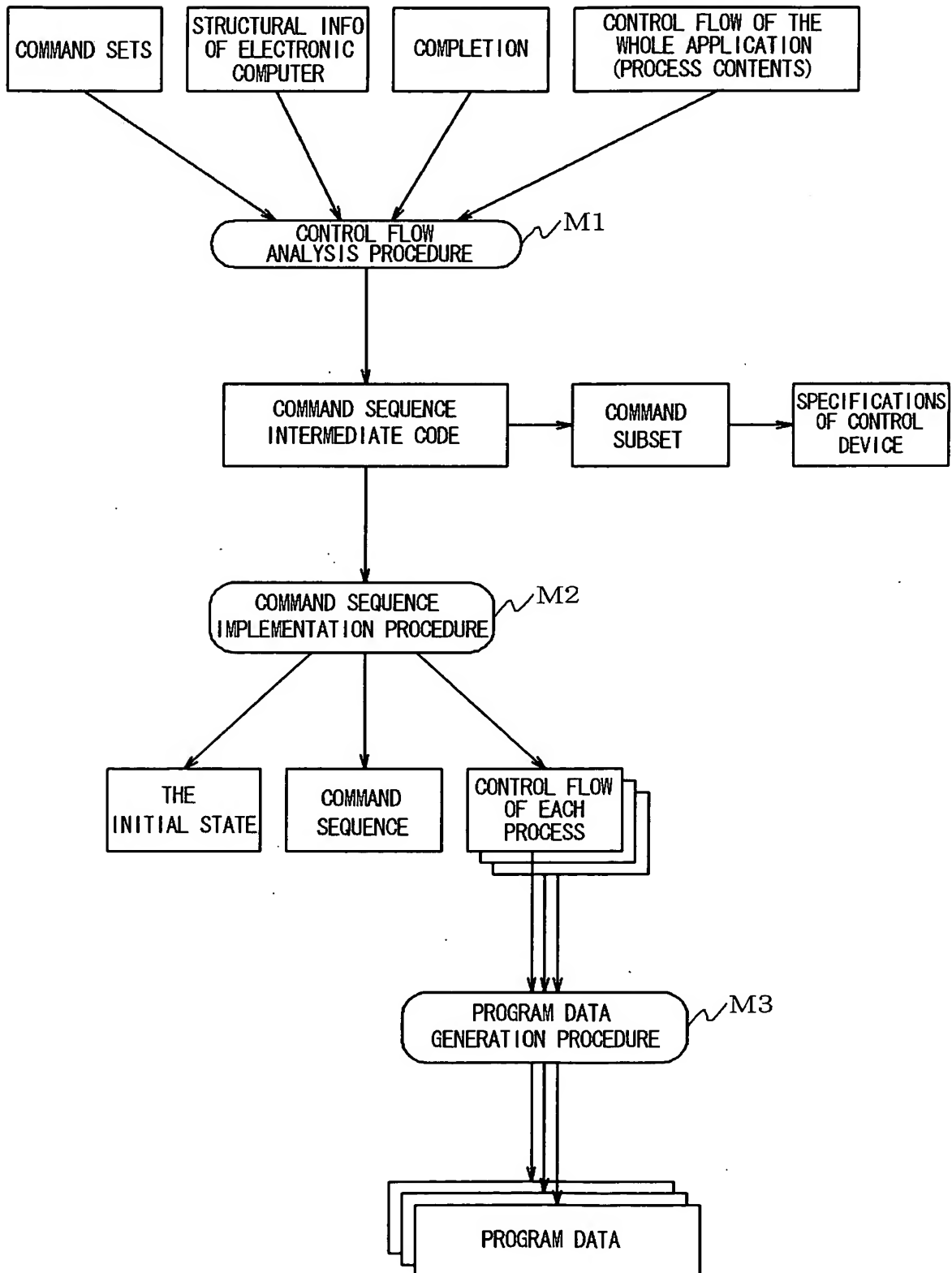


FIG . 30

PRIOR ART

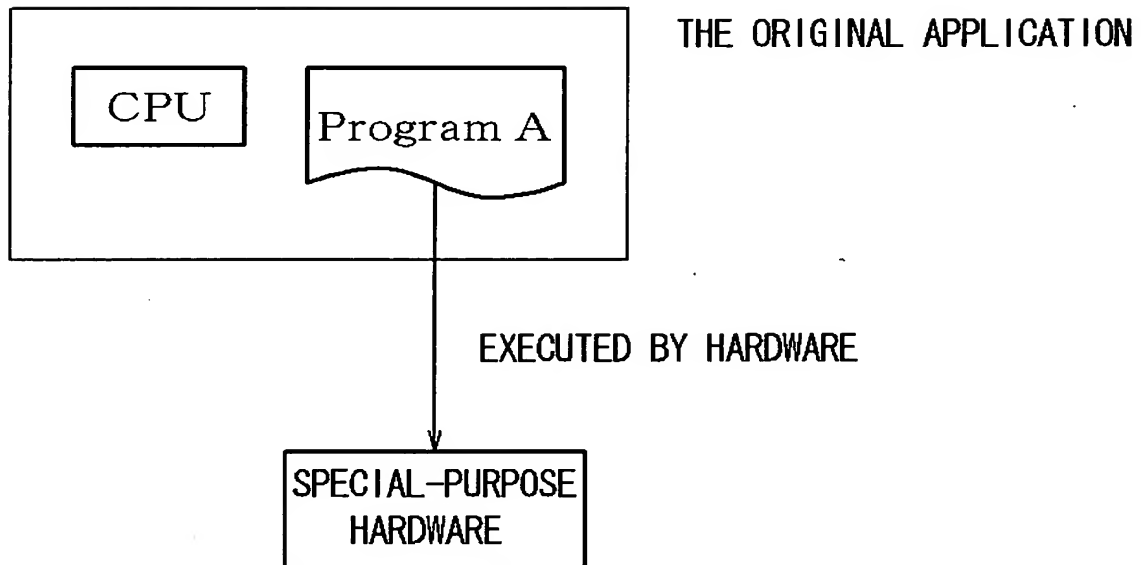


FIG . 31

PRIOR ART

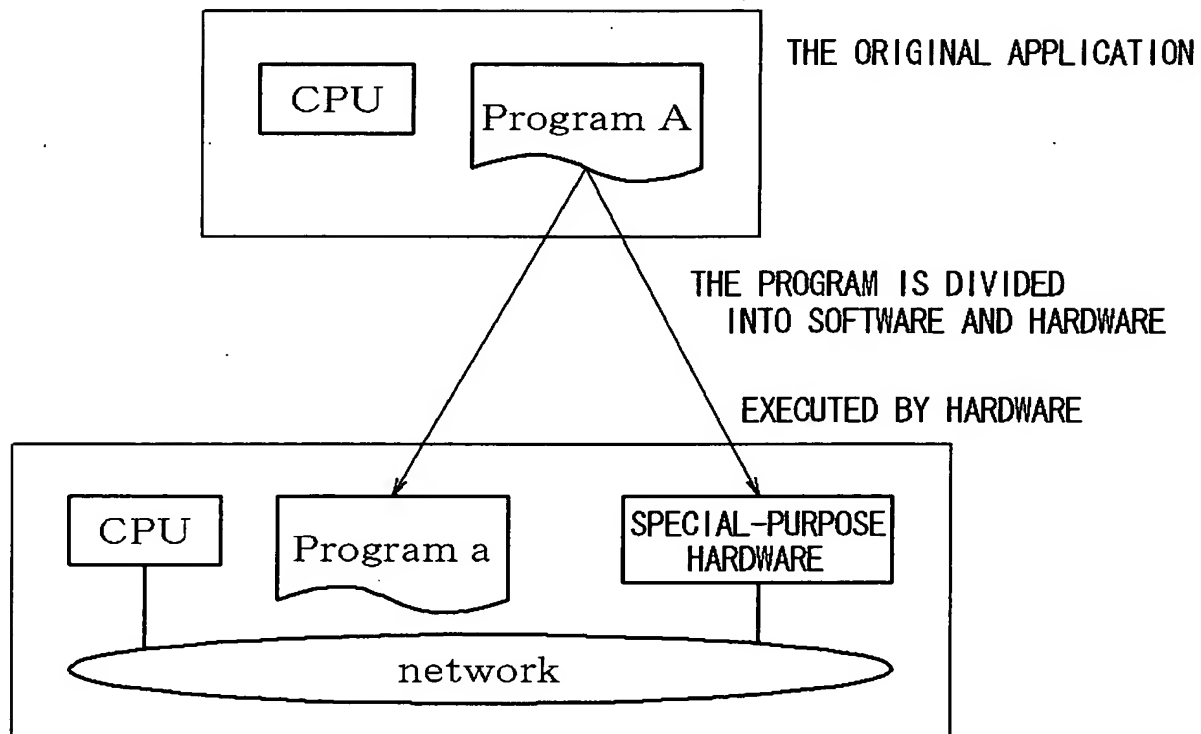


FIG . 32

PRIOR ART

